Trajectory Tracking Control of Quadrotor Unmanned Aerial Vehicle Using Sliding Mode Controller with the Presence of Gaussian Disturbance

Biruk Tadesse Nadew(✉), Asrat Mulatu Beyene, Beza Nekatibeb, and Mulugeta Debebe

Addis Ababa Science and Technology University, Addis Ababa, Ethiopia
{biruk.tadesse,asrat.mulatu,beza.nekatibeb, mulugeta.debebe}@aastu.edu.et

Abstract. In this research paper, trajectory following control of the quadrotor unmanned aerial vehicle (UAV) is carried out using the sliding mode controller (SMC) to control the position and attitude of the quadrotor, including the impact of disturbances. Due to their low cost, simplicity configuration of the structure, ability to hover, maneuverability, vertical take-off, and landing (VTOL) capability, and the ability to accomplished a variety of tasks, Quadrotor UAV has become an increasingly common research subject in recent years. The complex model of the quadrotor, including aerodynamic effects, was derived using Newton-Euler formalization. Here, SMC controllers were designed to track reference trajectories for the nonlinear model. The robustness and reliability of the controller are checked by applying external random disturbances to the quadrotor's position and attitude. Finally, by using MATLAB /Simulink, the action of the quadrotor/quadcopter under the suggested control scheme has been implemented. The results show that all control system becomes stable and robust in terms of disturbance rejection.

Keywords: UAV · Quadrotor · SMC · Dynamic modeling

1 Introduction

The maneuverability, low cost, structural simplicity, ability to flying and hovering, vertical take-off and landing capability, and ability to perform a different type of tasks in aerospace and control engineering, quadcopter UAV has become a popular research subject in recent years [1]. The various uses, such as video production, inspection, monitoring, law enforcement, rescue and search, and many others, inspire this study. UAV flights can be operated either fully autonomously by the onboard computers or by a certain degree of remote control by an operator on the ground or in a separate vehicle [2, 3].

M. A. Delele et al. (Eds.): ICAST 2020, LNICST 384, pp. 335–350, 2021.
https://doi.org/10.1007/978-3-030-80621-7_24

The analysis of the control design and simulation of autonomous drones has been approached in numerous ways by various literary works. The first dynamic quadrotor model was adopted via the Lagrangian approach. It was a partial differential equation for the quadcopter dynamic systems, that had been derived from the Lagrangian equation of the model [3, 4].

As discussed in [5, 6], it has been shown that it is possible to use linear control techniques to control the quadrotor UAV by linearizing the dynamics around the operating stage, commonly selected to be the hover. In this case, several unmodeled dynamics were overlooked and left out of the system during linearization, leading to a lack of controller efficiency. Backstepping [7, 8], sliding mode controller (SMC) [8, 9], and the feedback linearization [10, 11] were shown to be efficient for quadrotor control under these nonlinear control methods. Nevertheless, without understanding aerodynamic drag torque and force, gyroscopic torque, and unexpended environmental distributions, several of those experiments were investigated.

In [12], Wang et al. for a quadrotor developed a hierarchical system of time - varying adaptive control to follows 3D reference trajectory influenced by payload and time-varying disruption of wind raft variance. The concern was that the researcher had poorer control efficiency because of the neglect of the gyroscopic effect of a quadrotor and its rotors. [13], Bouzid et al. experimented with online disruption compensation using the model-free control (MFC) concept to deal with the uncertain component of the device (i.e., unmodelled dynamics, disruptions, etc.) on "trajectory tracking control of quadrotor UAV". The result obtained was fine, but rotor and quadrotor moment impacts were not taken into account in the work. The established nonlinear and complex dynamic model of the quadcopter is on the basis of a model of torque and thrust obtained from static thrust experiments with constant thrust and torque coefficients. As the vehicle undertakes complex maneuvers requiring substantial displacement and speeds [4–8]. Such a model is no longer true. In addition to excluding the disruption effect, the other issue is that most studies ignore the aerodynamic impact, the gyroscopic moment that exists because of the quadrotor body and its rotors [5, 6]. This affects the efficiency of trajectory tracking being decreased.

This paper discusses a thrust model that integrates the mediated momentum effects associated with the rotor and quadrotor body. Also, the aerodynamic drag force and torque are considered in the model. And Gaussian disturbance is added to the system angular and translational position and then, by using a sliding mode controller achieved better trajectory performance and fast disturbance rejection capability.

2 Mathematical Modelling of Quadrotor UAV

The system has all body frames, and inertial or earth frames F_b (x^b; y^b; z^b) and F_i (x^i; y^i; z^i) respectively. And the model split into rotational and translational coordinates, naturally [14].

$$\xi = (x;\ y;\ z) \in R^3, \quad \eta = (\phi;\ \theta; \psi) \in R^3 \tag{1}$$

Where x, y, and z are translational position for x, y, and z, dynamics respectively; and ϕ, θ, ψ, are the angular position of the roll, pitch, and yaw dynamics respectively. Using

Newton-Euler formalism, the following model equation rotational (Euler dynamics) and translational dynamics have been derived.

$$\left.\begin{array}{l} \ddot{\phi} = \frac{1}{I_x}\left(U_2 - k_4\dot{\phi}^2 - J_r\Omega_r\dot{\theta} + (Iy - Iz)\dot{\theta}\dot{\psi}\right) \\ \ddot{\theta} = \frac{1}{I_y}\left(U_3 - k_5\dot{\theta}^2 + J_r\Omega_r\dot{\phi} + (Iz - Ix)\dot{\phi}\dot{\psi}\right) \\ \ddot{\psi} = \frac{1}{I_z}\left(U_4 - k_6\dot{\psi}^2 + (Ix - Iy)\dot{\theta}\dot{\phi}\right) \\ \ddot{x} = \frac{1}{m}\left(\left(c_\phi s_\theta c_\psi + s_\phi s_\psi\right)U_1 - k_1\dot{x}\right) \\ \ddot{y} = \frac{1}{m}\left(\left(c_\phi s_\theta s_\psi - s_\phi c_\psi\right)U_1 - k_2\dot{y}\right) \\ \ddot{z} = \frac{1}{m}\left(\left(c_\phi c_\theta\right)U_1 - k_3\dot{z}\right) - g \end{array}\right\} \tag{2}$$

$$\dot{X} = f(X, U) \tag{3}$$

Where $\ddot{\phi}$, $\ddot{\theta}$, $\ddot{\psi}$ are the angular acceleration; $\dot{\phi}$, $\dot{\theta}$, $\dot{\psi}$, angular speed for translational roll, pitch, and yaw dynamics respectively. And $\ddot{x}$, $\ddot{y}$, $\ddot{z}$ are acceleration; $\dot{x}$, $\dot{y}$, $\dot{z}$ are velocity/speed vector for translational x, y, and z, dynamics respectively; m is mass quadrotor, g is gravitational acceleration; $[U_1, U_2, U_3, U_4]^T$ are the lift force, tilt (roll) torque, pitch torque, and yaw torque, and these are called control input vector for altitude, roll, pitch, and yaw, dynamics respectively. I_x, I_y, and I_z denote the moment of inertia of a quadrotor's x-axis, y-axis, and z-axis dynamics, respectively. Jr denotes the vertical or z-axis inertia of the rotors. $[k_1, k_2, k_3, k_4, k_5, k_6]$ are aerodynamic drag torque and force coefficients. The control inputs are assigned as follows:

$$\begin{array}{c} U1 = F_i = \sum_{i=1}^{4} F_i = b\sum_{i=1}^{4} w_i^2 = F_1 + F_2 + F_3 + F_4 = F \\ U2 = \tau_\phi = \mathcal{L}(F_4 - F_2) = \mathcal{L}b(w_4^2 - w_2^2) \\ U3 = \tau_\theta = \mathcal{L}(F_3 - F_1) = \mathcal{L}b(w_3^2 - w_1^2) \\ U4 = \tau_\psi = c(F_1 - F_2 + F_3 - F_4) = d(w_1^2 - w_2^2 + w_3^2 - w_4^2); \end{array} \tag{4}$$

Where Fi is the thrust force generated by the i^{th} rotors, w_i is the speed of the i^{th} rotors; d, b, $\mathcal{L}$ are thrust factor, drag factor, and length of quadrotor arm respectively.

X is the State vector, is

$$X = \left[\phi, \dot{\phi}, \theta, \dot{\theta}, \psi, \dot{\psi}, x, \dot{x}, y, \dot{y}, z, \dot{z}\right]^T \in R^{12} \tag{5}$$

And it can be written as

$$X = [x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}, x_{11}, x_{12}]^T \in R^{12} \tag{6}$$

$$U = [U_1, U_2, U_3, U_4]^T \tag{7}$$

The interaction between dynamics of positions (x, y), altitude (z) and rotational (ϕ, θ, ψ) are represented by the state-space model of the studied quadrotor is obtained as follows:

$$\begin{gathered}\dot{x}_1 = x_2, \dot{x}_2 = c_1x_4x_6 + c_2x_2^2 + c_3\Omega_r x_4 + b_1U_2\\ \dot{x}_3 = x_4, \dot{x}_4 = c_4x_2x_6 + c_5x_4^2 + c_6\Omega_r x_2 + b_2U_3\\ \dot{x}_5 = x_6, \dot{x}_6 = c_7x_2x_4 + c_8x_6^2 + b_3U_4\\ \dot{x}_7 = x_8, \dot{x}_8 = c_9x_8 + \frac{1}{m}(c_\phi s_\theta s_\psi + s_\phi s_\psi)U_1\\ \dot{x}_9 = x_{10}, \dot{x}_{10} = c_{10}x_{10} + \frac{1}{m}(c_\phi s_\theta s_\psi - s_\phi s_\psi)U_1\\ \dot{x}_{11} = x_{12}, \dot{x}_{12} = c_{11}x_{12} + \frac{1}{m}(c_\phi c_\theta)U_1 - g\end{gathered} \tag{8}$$

$$U_x = \left(c_\phi s_\theta c_\psi + s_\phi s_\psi\right) = c_{x_1}s_{x_3}c_{x_5} + s_{x_1}s_{x_5}$$

$$U_y = \left(c_\phi s_\theta c_\psi - s_\phi s_\psi\right) = c_{x_1}s_{x_3}s_{x_5} - s_{x_1}c_{x_5} \tag{9}$$

Where,

$c_1 = \frac{I_y - I_z}{I_x}$; $c_2 = -\frac{k_4}{I_x}$; $c_3 = -\frac{J_r}{I_x}$; $c_4 = \frac{I_z - I_x}{I_y}$; $c_5 = -\frac{k_5}{I_y}$; $c_6 = \frac{J_r}{I_y}$; $c_7 = \frac{I_x - I_y}{I_z}$ Ux
$c_8 = -\frac{k_6}{I_z}$; $c_9 = -\frac{k_1}{m}$; $c_{10} = -\frac{k_2}{m}$; $c_{11} = -\frac{k_3}{m}$; $b_1 = \frac{1}{I_x}$; $b_2 = \frac{1}{I_y}$; $b_3 = \frac{1}{I_z}$
and Uy are used to derive the reference trajectory for phi and theta (ϕ_d, θ_d) from the simulated loop (x and y dynamics). Therefore, the desired roll and pitch paths are derived from the position controllers from Eq. (9).

$$U_x = \left(c_{\phi_d}s_{\theta_d}c_\psi - s_{\phi_d}c_\psi\right) \text{ and } U_y = \left(c_{\phi_d}s_{\theta_d}s_\psi - s_{\phi_d}c_\psi\right) \tag{10}$$

$$\begin{bmatrix}\phi_d\\ \theta_d\end{bmatrix} = \begin{bmatrix}\arcsin\left(U_x s_\psi - U_y c_\psi\right)\\ \arcsin\left(\left(U_x c_\psi - U_y s_\psi\right)/c_{\phi_d}\right)\end{bmatrix} \tag{11}$$

3 SMC Design for the Nonlinear Dynamic Model

Sliding mode control (SMC) is a form of Variable Structure Control (VSC) which is performed by pulling the trajectories of the system state to the set-point surface called a sliding surface. The nature of the path of the state is separated into two parts [21]. This is named “the reaching stage” from the initial state until the intersection with the sliding surface. This direction is called “the sliding process” from the intersection with the sliding surface until the origin. Slotine and Li suggested the universal form of the equation in [15–20, 22].

$$S(x) = \left(\lambda_x + \frac{d}{dt}\right)^{f-1} e(x) \tag{12}$$

Where x is the control state or variable vector, e(x) is tracking deviation expressed as $x_d - x$, λ_x is a constant greater than zero that describes the surface dynamics and f is the relative degree of the sliding mode controller Entry conditions to the sliding surface were used for the Lyapunov-based method. This makes a positive scalar function, referred to below in Eq. (13), called the Lyapunov chosen function. The control rule that fulfills this function was selected for the dynamic state variables and selected [22]:

$$\dot{V} < 0, \text{ where } V > 0 \tag{13}$$

For selecting the Lyapunov stable function there is no universal principle. In this case the positive Lyapunov candidate function for each single dynamics systems the suitable function is chosen as:

$$V = \frac{1}{2}s^2 \tag{14}$$

Clearly seen that it is a positive definite, and its derivative $\dot{V} < 0$ (i.e., $\dot{V} = s\dot{s} < 0$). The aim is to attract the paths of the machine state to hit the sliding or switching surface and, considering the existence of ambiguity, remain on it. Tracking error is the deviation between the desired result and the state's exact expected values, defined as follows:

$$e_{i+1} = \dot{e}_1;\ e_i = x_{id} - x_i,\ (i = 1, 2 \ldots 6) \tag{15}$$

Based on the reaching low, "constant reaching low" the derivative of the sliding surface meets that $\dot{V} = s\dot{s} < 0$ condition given by [23] as

$$\dot{s}_i = -K_i \operatorname{sgn}(s_i) \tag{16}$$

Thus $K_i > 0$, represents a constant rate.

$$\operatorname{sgn}(s_i) = \begin{cases} 1, & s_i > 0 \\ 0, & s_i = 0 \\ -1, & s_i < 0 \end{cases} \tag{17}$$

K_i's value is too little; it will take too long to hit the period. On the other hand, too much Ki value will produce extreme chattering, so the success of SMC will determine the option of K_i value. Equation (14) is used to choose the SMC sliding surface feature so that the sliding surface can be chosen on the basis of tracking deviation, as seen below:

$$s_1 = \lambda_1\phi_e + \dot{\phi}_e, \quad \phi_e = \phi_d - \phi = e_1 \tag{18}$$

$$s_2 = \lambda_2\theta_e + \dot{\theta}_e, \quad \theta_e = \theta_d - \theta = e_2 \tag{19}$$

$$s_3 = \lambda_3\psi_e + \dot{\psi}_e, \quad \psi_e = \psi_d - \psi = e_3 \tag{20}$$

$$s_6 = \lambda_6 z_e + \dot{z}_e, \quad z_e = z_d - z = e_6 \tag{21}$$

$$s_x = \lambda_4 x_e + \dot{x}_e, \quad x_e = x_d - x = e_4 \tag{22}$$

$$s_y = \lambda_5 y_e + \dot{y}_e, \quad y_e = y_d - y = e_5 \tag{23}$$

Where, $\lambda_{1\ldots6}$ must satisfy the Hurwitz stability condition or greater than zero.

Equation (18), (19), (20), (21), (22), and (23), are the sliding surface & its tracking deviation for roll dynamics; for the pitch dynamics; for the yaw dynamics; for altitude dynamics, for x-dynamics, for y-dynamics respectively. The nonlinear dynamic model is split into subsystems called internal and external dynamics. The position dynamics of the internal system (X and Y) produces the optimal angular position called roll and pitch (ϕ_d and Θ_d). The rest subsystem is the altitude, roll, pitch, and yaw dynamics of the external system (z, ϕ, Θ, ψ) [14, 16–18].

In the case of internal dynamics, it influences the rotation of the quadcopter. However, external interference is made, which in turn affects the movement of the quadcopter. So, to stabilize this problem, the SMC is used.

3.1 SMC Design for the Dynamics of Quadrotor

The tracking deviation or error, and the switching surface for tilt (roll) angle determined in Eq. (18) the 1st derivative of the sliding surface is

$$\dot{s}_1 = \lambda_1\dot{e}_1 + \ddot{e}_1 = \lambda_1\dot{\phi}_e + \ddot{\phi}_e = \lambda_1(\dot{\phi}_d - \dot{\phi}) + \ddot{\phi}_d - \ddot{\phi}; \tag{24}$$

From Eq. (16) $\dot{s}_1 = -K_1 sgn(s_1)$ & Eq. (2): $\ddot{\phi} = \dot{x}_2 = c_1x_4x_6 + c_2x_2^2 + c_3\Omega_r x_4 + b_1U_2$ equating with Eq. (24 c) and then, U_2 an be calculated as follows:

$$U_2 = \frac{1}{b_1}\left[\ddot{\phi}_d - c_1x_4x_6 + c_2x_2^2 + c_3\Omega_r x_4 + \lambda_1(\dot{\phi}_d - \dot{\phi}) - \dot{s}_1\right], \quad \text{or}$$

$$U_2 = \frac{1}{b_1}\left[\ddot{x}_{1d} - c_1x_4x_6 - c_2x_2^2 - c_3\Omega_r x_4 + \lambda_1\dot{e}_1 + K_1 sgn(s_1)\right] \tag{25}$$

Where, $\dot{e}_1 = \left(\dot{\phi}_d - \dot{\phi}\right) = \dot{\phi}_e$. This design approach is repeating for all later systems. The controller becomes:

$$U_3 = \frac{1}{b_2}\left[\ddot{x}_{3d} - c_4x_2x_6 - c_5x_4^2 - c_6\Omega_r x_2 + \lambda_2\dot{e}_2 + K_2 sgn(s_2)\right] \tag{26}$$

$$U_4 = \frac{1}{b_3}\left[\ddot{x}_{5d} - c_7x_2x_4 - c_8x_6^2 + \lambda_3\dot{e}_3 + K_3 sgn(s_3)\right] \tag{27}$$

$$U_1 = \frac{m}{c_{x1}c_{x3}}\left[\ddot{z}_d - c_{11}x_{12} + g + \lambda_6\dot{e}_6 + K_6 sgn(s_6)\right] \tag{28}$$

The same algorism is used to drive the control laws for U_x and U_y to stabilize the positions X and Y dynamics of the Quadcopter, respectively. 1st derivative of sliding surface for x-dynamics is calculated as:

$$\dot{s}_4 = \lambda_4\dot{e}_4 + \ddot{e}_4 = \lambda_4\dot{x}_e + \ddot{x}_e = \lambda_4(\dot{x}_d - \dot{x}) + \ddot{x}_d - \ddot{x}, \;\; \dot{s}_4 = -K_4 sgn(s_4) \tag{29}$$

The nonlinear dynamic modeling in Eq. (1) for the translational X dynamics are

$$\ddot{x} = \dot{x}_8 = \ddot{x}_7 = c_9x_8 + \frac{1}{m}(c_\phi s_\theta c_\psi + s_\phi s_\psi)U_1$$

Where, $u_x = (c_\phi s_\theta c_\psi + s_\phi s_\psi) = c_{x_1} s_{x_3} c_{x_5} + s_{x_1} s_{x_5}$

Therefore, the x-dynamics acceleration became $\ddot{x} = c_9 x_8 + \frac{1}{m} U_x U_1$, Equating from Eq. (29), the SM control input U_x for desired X position can be obtained as:

Therefore, the x-dynamics acceleration became $\ddot{x} = c_9 x_8 + \frac{1}{m} U_x U_1$, Equating from Eq. (28), the SM control input U_x for desired X position can be obtained as:

$$U_x = \left\{ \frac{m}{U_1} \left[\ddot{x}_{7d} - c_9 x_8 + \lambda_4 \dot{e}_4 + K_4 \mathrm{sgn}(s_4) \right] \right\} \tag{30}$$

The same approaches for y-dynamics the controller becomes

$$U_y = \frac{m}{U_1} \left[\ddot{x}_{9d} - c_{10} x_{10} + \lambda_5 \dot{e}_5 + K_5 \mathrm{sgn}(s_5) \right] \tag{31}$$

Figure 1 shows the overall system's SIMULINK block diagram. It includes controller of altitude (Z-dynamics), X-position, Y-position, heading (yaw), attitude roll, and attitude pitch dynamics control sub-system blocks. The parameter used for simulation are listed in Tables 1 and 2.

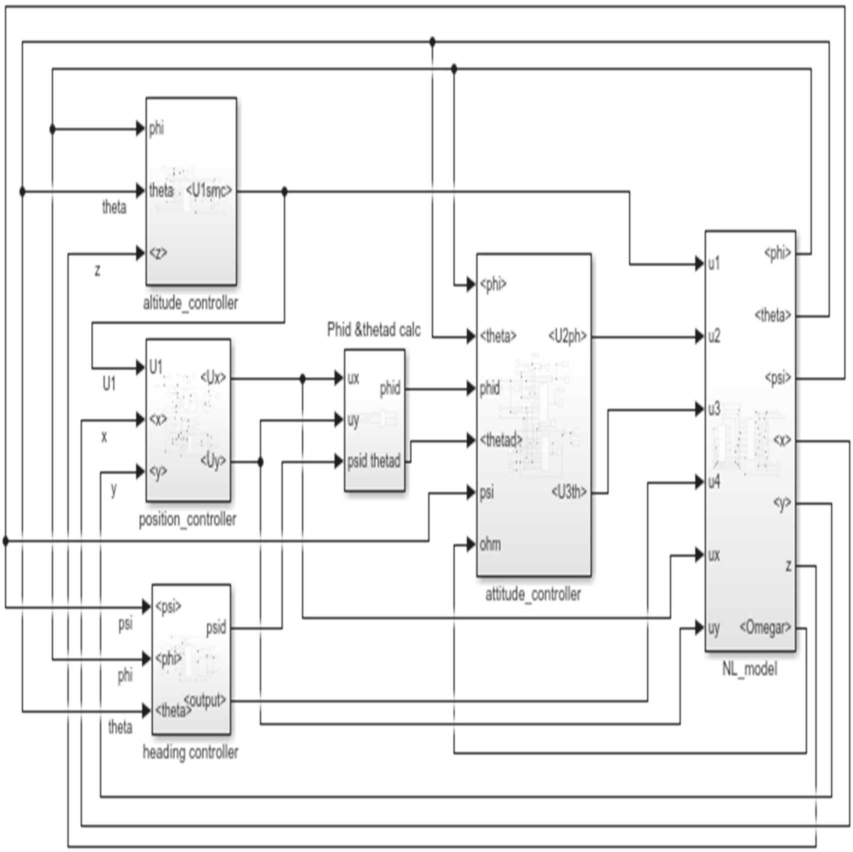


Fig. 1. Overall control system SIMULINK block diagram

Table 1. Parameters used for SMC simulation purpose

Dynamics	K_i (accepted constant)	λ_i	ρ (boundary)
Roll-attitude	224.5	25.05	0.00099
Pitch-attitude	224.5	25.05	0.00099
Yaw-heading	223	15.05	0.00099
X - position	334	15.05	0.00099
Y - position	334	15.05	0.00099
Z - altitude	224.5	15.05	0.00099

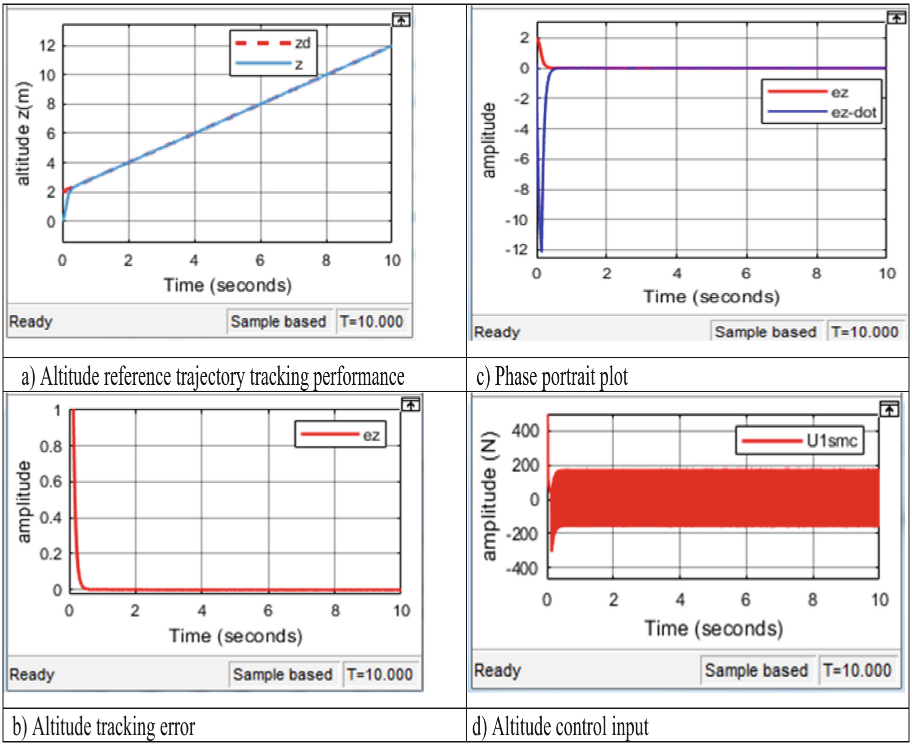
Table 2. Parameters used for the quadrotor model simulation

Parameter	Value and unit
Coefficient of lift (b)	$29.8*10^{-4}$ Ns^2 rad^{-2}
Coefficient of drag (d)	$33*10^{-6}$ Nms^2 rad^{-2}
Total mass (m)	$8.23*10^{-1}$ kg
Length of arm (l)	0.350 m
Inertia for motor (J_r)	$28.4*10^{-4}$ kg m^2
Aerodynamic friction coeffs ($K_{1,2,3}$)	$37.29*10^{-2}$
Inertia moment for quadrotor (Ix,y,z)	$10^{-2}*\{0.5, 0.5, 0.1\}$ in (kg m^2) resp.
aerodynamic drag coeffs for translational ($K_{4,5,6}$)	$55.6*10^{-3}$
Gravity (g)	9.81 m/s^2

4 Simulation Results and Analysis

4.1 Simulation Result of the Control System Without Disturbance

In this part, the stability and consistency of the proposed control scheme are examined without and with disruption. The overall control system which is designed using MATLAB/Simulink. It includes the altitude (z), position (x, y), attitude (ϕ, θ), and heading (ψ) control sub-system. Figure 2 illustrates tracking performance, tracking error, phase portrait, and altitude control input of the altitude dynamics.

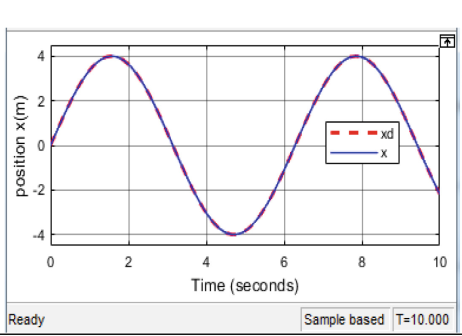


a) Altitude reference trajectory tracking performance
c) Phase portrait plot
b) Altitude tracking error
d) Altitude control input

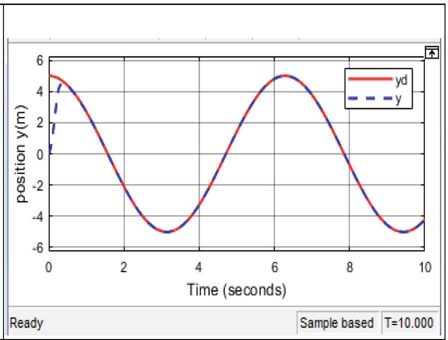
Fig. 2. a) Altitude reference trajectory tracking controller performance b) tracking error c) phase-plot d) control input of altitude SMC

Clearly seen from Fig. 2, the set-point path starts from two (m) after 0.35 s the real path tracks the set-point (desired) zd = (2 + t) and starts from 0.45 s, the tracking error is very small nearly 0.002 s. The phase plot shows that the control system error change is going to zero within 0.3 s. This implies the control system is stable.

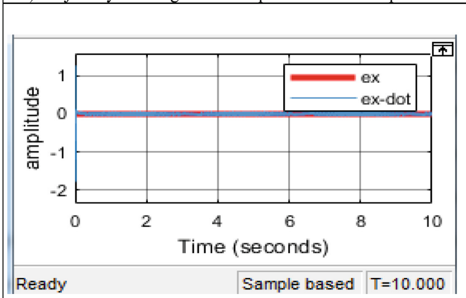
Figure 3 shows the simulation result of the SM controller to track the directions X and Y in the comparison trajectories. The real path (x) tracks the specified path (xd = 4*sin(t)) at 0 s. In the cases of Y position, the real trajectory is the optimal one (yd = 5*cos(t)) after 0.4 s. The result reveals that the suggested controller has accomplished its monitoring targets in both positions.



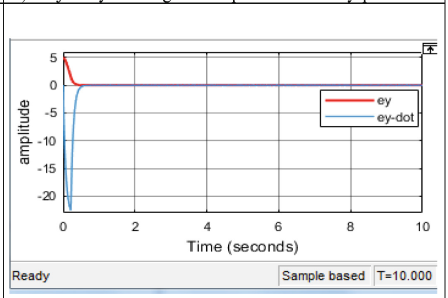
a) Trajectory tracking controller performance of x-position



b) Trajectory tracking control performance of y-position



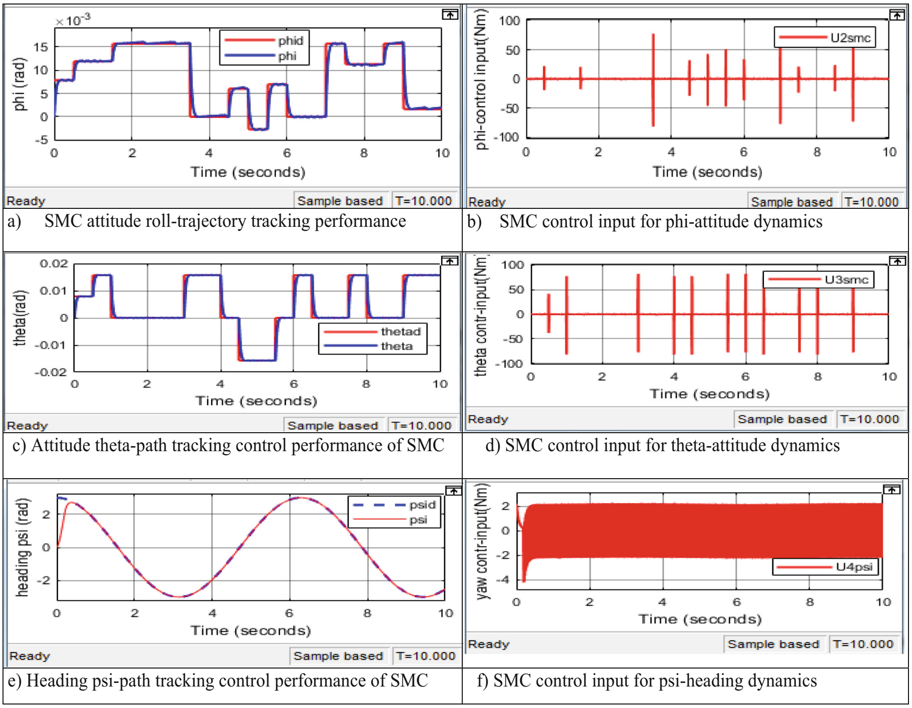
c) Phase portrait for position x dynamics



d) Phase portrait of the y dynamics position

Fig. 3. Trajectory tracking abilities of position and its phase portrait for x, y dynamics

Figure 4 shows the trajectory tracking performance and their SMC control input for roll, for pitch and, for yaw trajectory tracking. The roll and pitch dynamics are called the attitude of the quadrotor. Its dynamics are generated from the X and Y position dynamics. The controller performance objectives are achieved for the three rotational dynamics (i.e., Roll, Pitch, and Yaw) of the quadrotor.



a) SMC attitude roll-trajectory tracking performance

b) SMC control input for phi-attitude dynamics

c) Attitude theta-path tracking control performance of SMC

d) SMC control input for theta-attitude dynamics

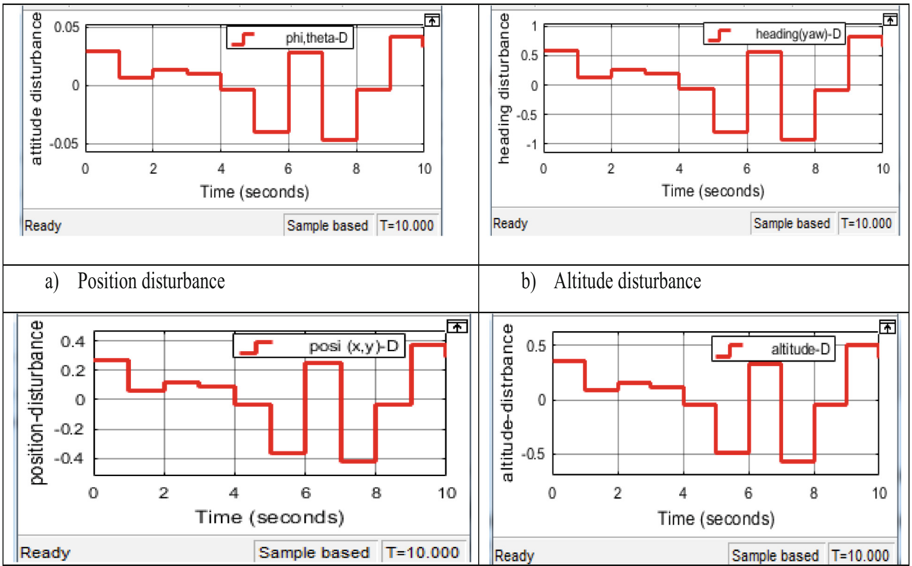
e) Heading psi-path tracking control performance of SMC

f) SMC control input for psi-heading dynamics

Fig. 4. Trajectory tracking performance and its SMC control input for rotational roll, pitch, and yaw dynamic

4.2 Simulation Results of the Proposed Control System with Gaussian Disturbance

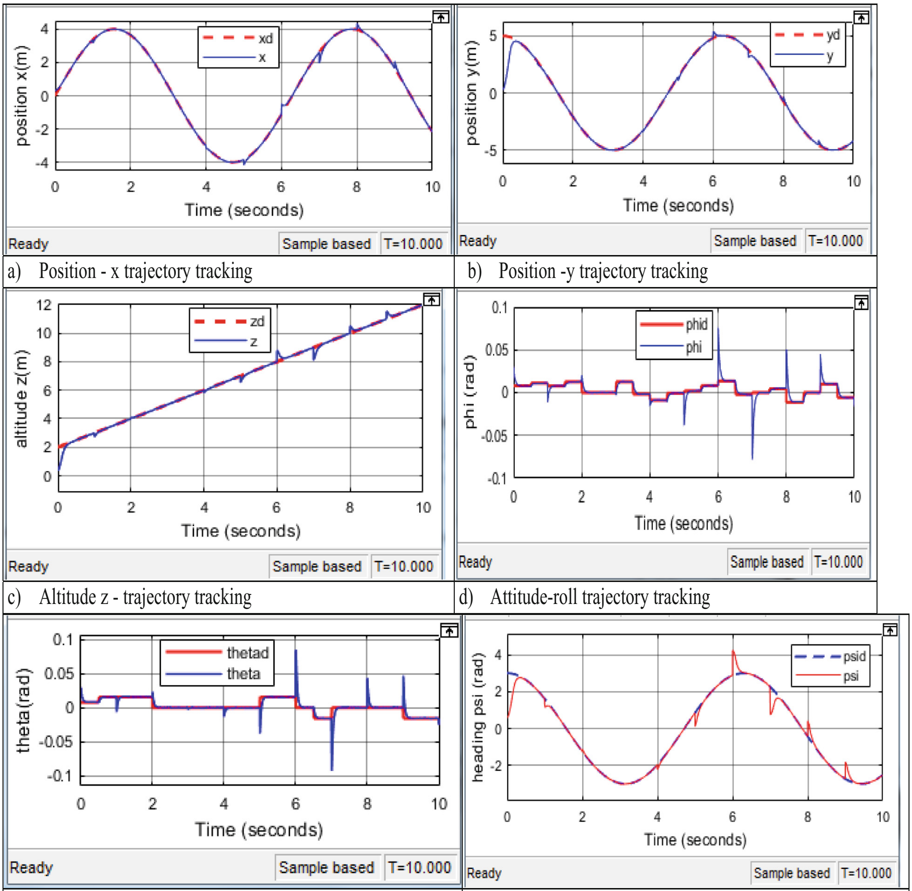
Here the designed control system is tested by adding external disturbance (it is called Gaussian disturbance) on the system. The simulation results in Fig. 5; illustrate the output of the Gaussian (normal) spontaneous disruption signal. Particularly, this disruption involves the variation of the air, the wind, and the rain effects that occur in the quadrotor's actual world.



a) Position disturbance
b) Altitude disturbance
c) Attitude disturbance
d) Heading (yaw) disturbance

Fig. 5. External random Gaussian disturbances that are added on position, altitude, attitude, and heading dynamics

Figure 6 demonstrate that the unknown disruptions enter the system unexpectedly, the proposed SM controller immediately takes the necessary action to change the system to be stable and refuse the effects of the added disturbance. The time taken by the controller to eliminate the Gaussian type of disturbance is less than 0.2 s for all the dynamics. The overall simulation results depict that in the cases of disturbance avoidance, the proposed SM controller scheme is robust and versatile.



a) Position - x trajectory tracking
b) Position -y trajectory tracking
c) Altitude z - trajectory tracking
d) Attitude-roll trajectory tracking
e) Attitude-pitch trajectory tracking
f) heading-yaw trajectory tracking

Fig. 6. Trajectory tracking performance of a) position-x, b) position-y, c) altitude, d) roll, e) pitch, and f) yaw controller with disturbance

4.3 3D-Helical Trajectory Tracking Performance

Here for translational (X, Y, and, Z) dynamics in space, the designed SMC approaches for three-dimensional helical trajectory following controllers were applied. It is visible from Fig. 7; effective tracking of the 3D-helical track was achieved. This shows a 3D-helical reference trajectory produced when the quadrotor initially flies from point (0, 0, 0) m after trajectories (4sin(t), 5cos(t), 2 + t) m in the directions of X, Y, and Z, respectively. This 3D-helical pathway simulation result illustrates the reliability of the controller and its robustness. That means the controller manages any type of motion of the quadrotor.

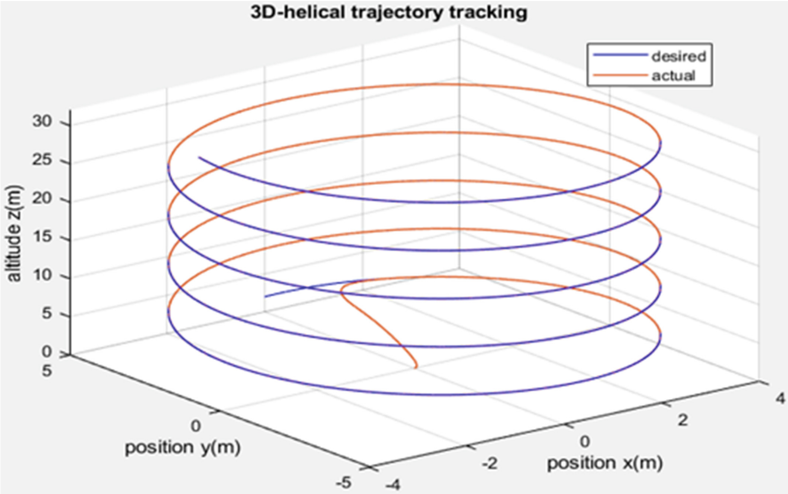


Fig. 7. SMC 3D-helical trajectory tracking for (4sin (t), 5cos (t), 2 + t) m

5 Conclusions and Future Works

5.1 Conclusion

The quadrotor (quadcopter) nonlinear dynamic model is obtained by using the Newton-Euler force-torque equation. Here there are translational and rotational (Euler) dynamics in the nonlinear quadcopter model. The effect of gyroscopic moment resulting from the quadcopter body and its rotor blade takes into account and the model also integrates aerodynamic drag force and torque. The sliding mode controller was designed for the dynamic model of the quadrotor. The simulation using MATLAB / Simulink was performed to verify the validity and performance of the proposed controller here.

The sliding mode controllers are used for monitoring the angular dynamics (attitude and heading) translational dynamics (position and altitude) of the quadcopter. The position (x, y), and altitude control system contains three states. The control inputs are strongly coupled and nonlinear for both input and output. Therefore, SMC is used to stabilize this type of nonlinear coupled systems. The result shows SMC makes the actual path to track the reference one within 0.5 s in the average of the overall system. And it performs better tracking ability with a minimum error. Many unknown and unmodelled disruptions influence its motion and the quadrotor's stability while the quadcopter is operating. To validate the robustness of the controller by guessing the nonlinear external disturbance within a certain range and adding it to the quadrotor output terminal. This impact of disruption was controlled effectively by the suggested controller. That means that the controller changes (takes a swift action) immediately and makes a stable track. Thus, in case of disturbance rejection, the planned control method is robust.

5.2 Future Works

In the dynamic model of the quadrotor, the controllers are derived by considering certain assumptions (such as the quadcopter configuration is rigid and symmetrical, the propeller blades are rigid, and the rotational system dynamics are bound in range to prevent singularities). The dynamic model and design control system without these considerations were obtained for further investigation. Unmodeled noise (chattering) effects may be minimized. However, another study might adjust the controller scheme to optimize using particle swarm (PSO) or generic algorithm (GA) or fuzzy logic tuning techniques. And the other direction of research is the higher-order and integral (HISMC) instead of SMC. This could increase its tracking efficiency for different trajectories.

References

1. Jategaonkar, R.: Flight Vehicle System Identification a Time Domain Methodology. American Institute of Aeronautics and Astronautics Inc., Reston (2015)
2. Norris, D.: Build Your Own Quadcopter. McGraw-Hill Education, New York (2014)
3. Vachtsevanos, K.P., Valavanis, G.J.: Handbook UAV. Springer, New York (2015). https://doi.org/10.1007/978-90-481-9707-1
4. Quan, Q.: Introduction to Multicopter Design and Control. Springer, Beijing (2017). https://doi.org/10.1007/978-981-10-3382-7
5. Bouabdallah, S., Noth, A., Siegwart, R.: PID vs LQ control techniques applied to an indoor micro quadrotor. In: International Conference on Intelligent Robots and Systems (2004)
6. Pounds, P., Mahony, R., Corke, P.: Modelling and control of a quadrotor. In: Australasian Conference on Robotics and Automation, Auckland, NZ (2006)
7. Madani, T., Benallegue, A.: Backstepping control for a quadrotor helicopter. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (2006)
8. Bouabdallah, S., Siegwart, R.: Backstepping and sliding-mode techniques applied to an indoor micro quadrotor. In: IEEE International Conference on Robotics and Automation, Barcelona, Spain (2005)
9. Xu, R., Ozguner, U.: Sliding mode control of a quadrotor helicopter. In: Proceedings of the 45th IEEE Conference on Decision and Control, San Diego, CA, USA (2006)
10. Lee, D., Kim, H.J., Sastry, S.: Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter. Int. J. Control Autom. Syst. **3**(7), 419–428 (2009)
11. Tengis, Ts., Batmunkh, A.: State feedback control simulation of quadcopter model. In: IEEE 2016 11th International Forum on Strategic Technology, Novosibirsk, Russia (2016)
12. Wang, C., Song, B., Huang, P., Tang, C.: Trajectory tracking control for quadrotor robot subject to payload variation and wind gust disturbance. J. Intell. Robot. Syst. (2016)
13. Bouzid, Y., Siguerdidjane, H., Bestaoui, Y., 3D trajectory tracking control of quadrotor UAV with on-line disturbance compensation. In: IEEE Conference on Control Technology and Applications, Kohala Coast, Hawaii, USA, August 2017
14. Rezoug, A., Hamerlain, M., Achour, Z., Tadjine, M.: Applied of an adaptive higher order sliding mode controller to quadrotor trajectory tracking. In: IEEE International Conference on Control System, Computing and Engineering, Penang, Malaysia, November 2015
15. Shtessel, Y., Edwards, C., Fridman, L., Levant, A.: Sliding Mode Control and Observation. Springler, Birkhäuser (2014). https://doi.org/10.1007/978-0-8176-4893-0
16. Johan From, P., Tommy Gravdahl, J., Ytterstad Pettersen, K.: Vehicle-Manipulator System. Springer, London (2014). https://doi.org/10.1007/978-1-4471-5463-1

17. Ghazbi, S.N., Aghli, Y., Alimohammadi, M.: Quadrotors unmanned aerial vehicles: a review. Int. J. Smart Sens. Intell. Syst. **9**(1) (2016)
18. Villanueva, A., Castillo-Toledo, B., Bayro-Corrochano, E.: Multi-mode flight sliding mode control system for a quadrotor. In: 2015 International Conference on Unmanned Aircraft Systems (ICUAS), Denver, Colorado, USA, June 2015
19. Yi, K., Gu, F., Yang, L., He, Y., Han, J.: Sliding mode control for a quadrotor slung load system. In: Proceedings of the 36th Chinese Control Conference, Dalian, China, 26–28 July 2017
20. Gherouat, O., Matouk, D., Hassam, A., Abdessemed, F.: Modeling and sliding mode control of a quadrotor unmanned aerial vehicle. J. Autom. Syst. Eng. **10**(3), 150–157 (2016)
21. Jiukin, L.: Sliding Mode Control Using MATLAB. Elsevier Inc., Beijing (2017)
22. Liu, J., Wang, X.: Advanced Sliding Mode Control for Mechanical Systems. Springer, Dordrecht (2014). https://doi.org/10.1007/978-3-642-20907-9
23. Derbel, N., Ghommam, J., Zhu, Q. (eds.): Applications of Sliding Mode Control. SSDC, vol. 79. Springer, Singapore (2017). https://doi.org/10.1007/978-981-10-2374-3